下载APP

02｜YUV & RGB：原来图像是这么丰富多彩的

2021-11-24 李江

《攻克视频技术》 课程介绍 >

讲述：李江

时长 18:15 大小 16.72M

你好，我是李江。今天我们一起来聊一聊图像的颜色空间。

图像的颜色空间是图像和视频技术里面的一个非常重要的知识点，在图像处理、视频编码等技术中你会经常遇到这个概念。

而至于它为什么重要，其实也很好理解。在现实世界中，我们的眼睛每天看到的颜色是千变万化的。为了能够更方便地表示和处理这些颜色，不同应用领域就建立了多种不同的颜色空间，主要包括 RGB 、YUV、CMYK 、 HSI 等（后面两种和这门课程没有关系，因此这里我们不再介绍）。

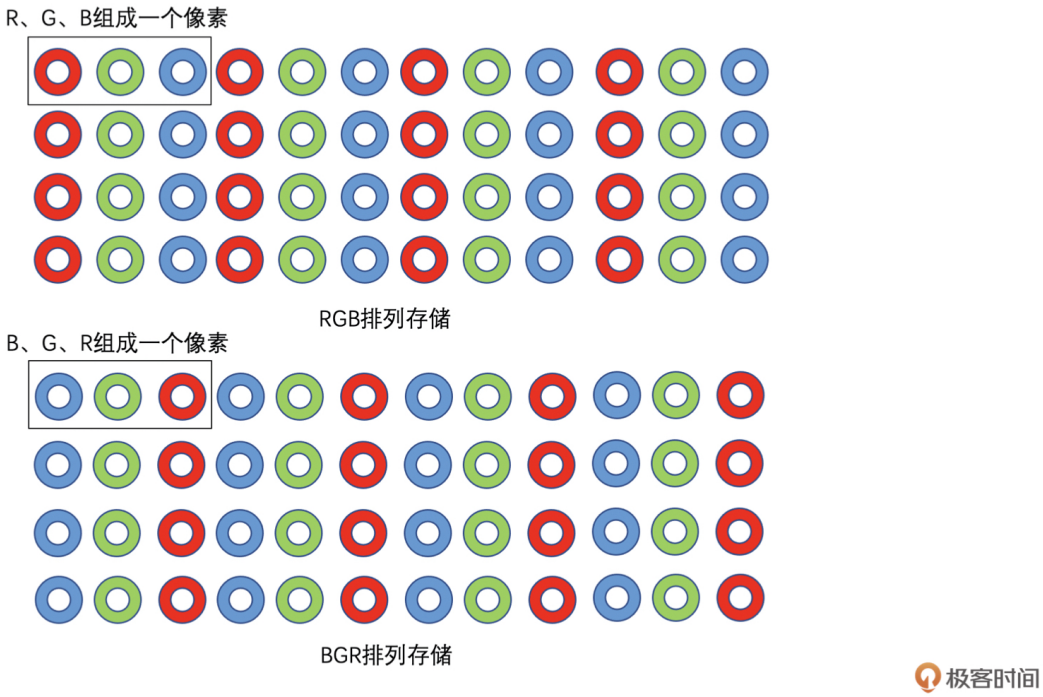
在视频技术中，我们经常碰到的颜色空间有两种：RGB 和 YUV，接下来我们就逐一看一看。

RGB

RGB 相对比较简单。顾名思义，它就是指图像的每一个像素都有 R、G、B 三个值。RGB 是我们平常遇到最多的一种图像颜色空间，比如摄像头采集的原始图像就是 RGB 图像，且显示器显示的图像也是 RGB 图像。

一般来说，**我们的 RGB 图像，每一个像素都是分别存储 R、G、B 三个值，且三个值依次排列存储**。比如对于一张 8bit 位深的 RGB 图，每个值占用一个字节。但是，需要注意的是 RGB 图像像素中 R、G、B 三个值并不一定是按 R、G、B 顺序排列的，也有可能是 B、G、R 顺序排列。

比如 OpenCV 就经常使用 BGR 的排列方式来存储图像。所以在存储和读取 RGB 图像的时候需要稍微注意一下。RGB 和 BGR 的存储方式如下图所示：



虽然 RGB 比较简单，同时在图像处理的时候也经常会用到。但是在视频领域，我们更多地是使用 YUV 颜色空间来表示图像的。这是因为 R、G、B 三个颜色是有相关性的，所以不太方便做图像压缩编码。那 YUV 究竟是怎么表示图像的呢？它又是如何存储在内存当中的呢？我们接下来就来揭开它的“面纱”。

YUV

YUV 跟 RGB 类似，也是一种颜色空间，但其种类会更多更复杂些，所以接下来我们会花大量的篇幅去讲解它。

YUV 最早主要是用于电视系统与模拟视频领域。现在视频领域基本都是使用 YUV 颜色空间。

跟 RGB 图像中 R、G、B 三个通道都跟色彩信息相关这种特点不同，**YUV 图像将亮度信息 Y 与色彩信息 U、V 分离开来。Y 表示亮度，是图像的总体轮廓，称之为 Y 分量。U、V 表示色度，主要描绘图像的色彩等信息，分别称为 U 分量和 V 分量**。这样一张图像如果没有了色度信息 U、V，只剩下亮度 Y，则依旧是一张图像，只不过是一张黑白图像。这种特点有什么好处呢？

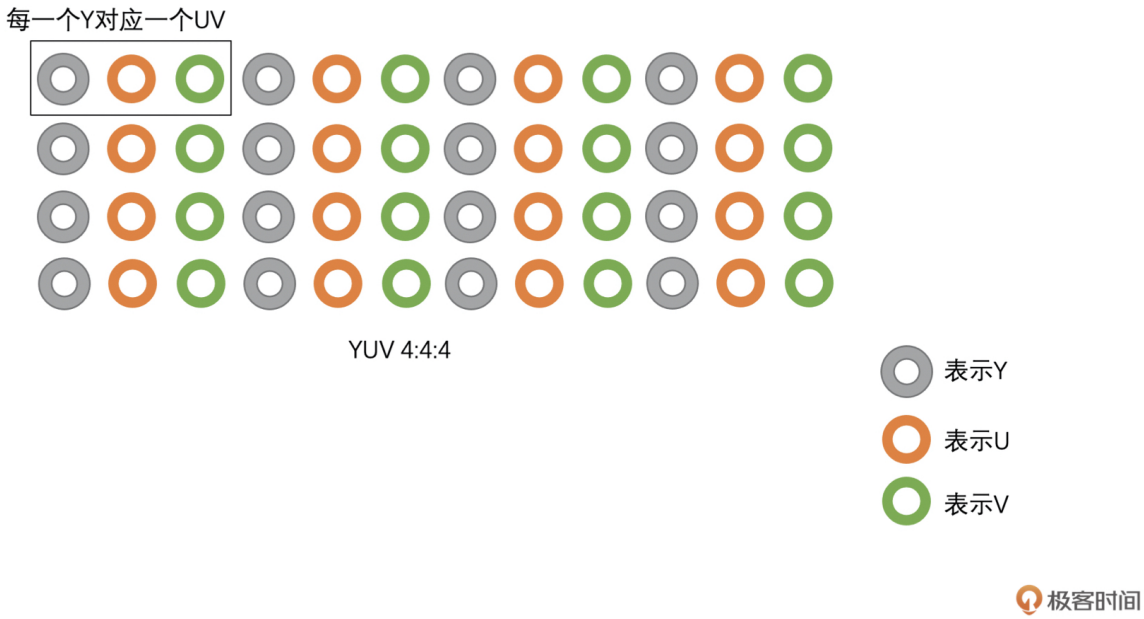
在以前，世界上只有黑白电视机，每一帧电视画面都是黑白的，没有色彩信息。当然黑白电视机也不支持显示彩色图像。后来随着技术的发展，出现了彩色电视机，每一帧画面都是有颜色信息的，那当然我们可以使用 RGB、YUV 等颜色空间来表示一帧图像。

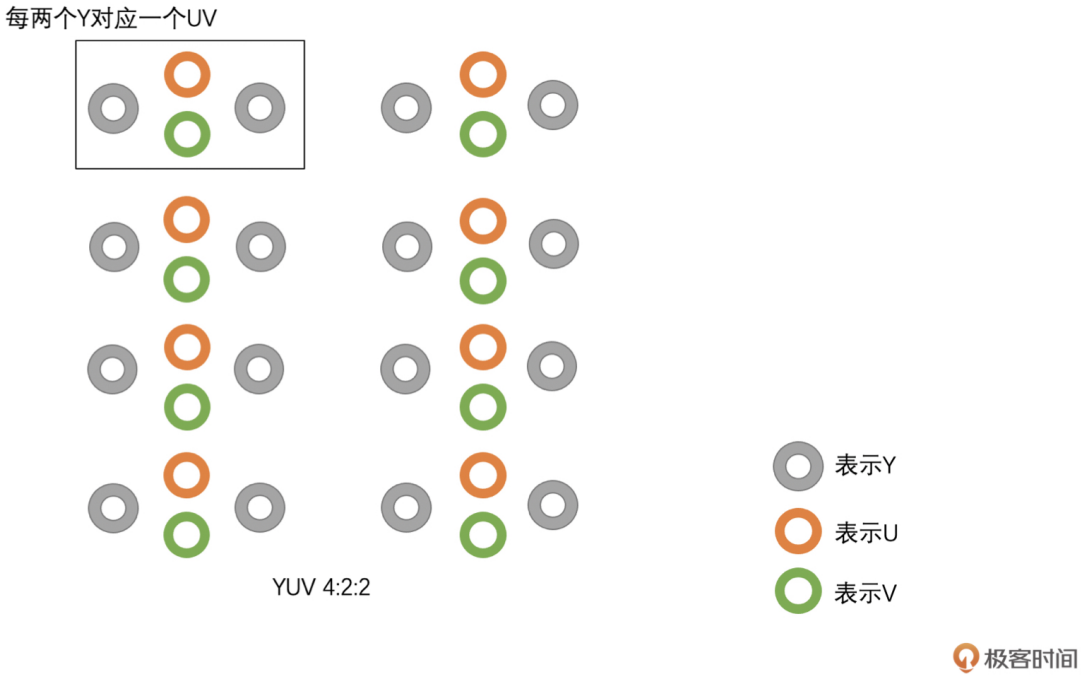
但是考虑到兼容老的黑白电视机，如果使用 RGB 表示图像，那么黑白电视机就没办法播放。这是因为 R、G、B 三个通道都是彩色的，而 Y、U、V 就可以。因为黑白电视机可以使用 Y 分量，Y 分量就是黑白图像，而且包含了图像的总体轮廓信息，只是没有色彩信息而已。

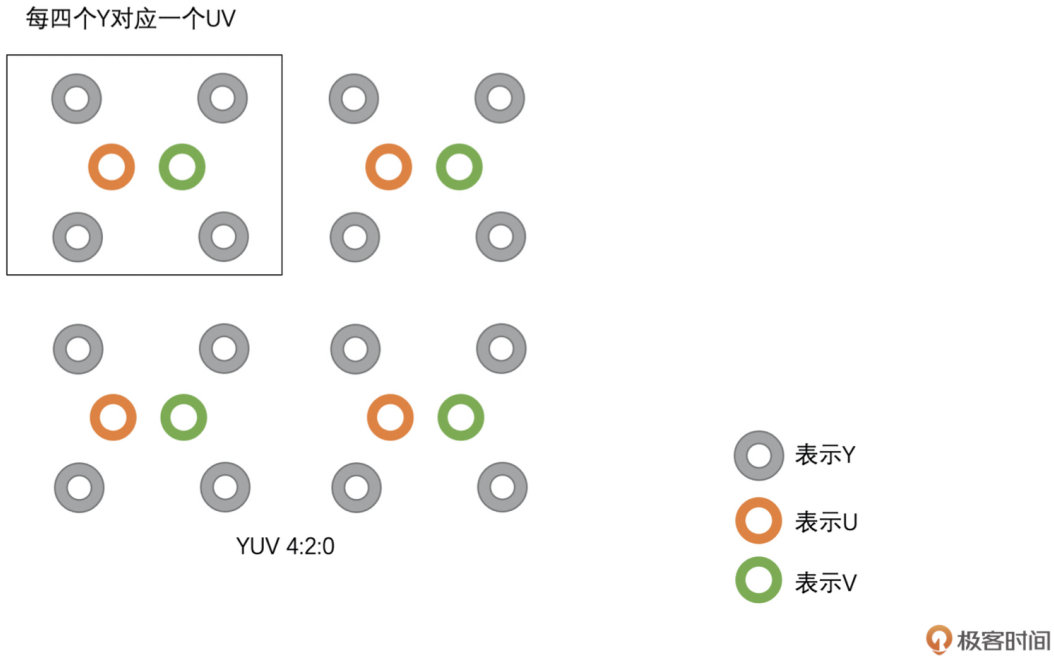
好了，讲了 YUV 的起源之后，我们来讲讲 YUV 颜色空间最重要的知识点，那就是 YUV 的类型和存储方式。

YUV 主要分为 YUV 4:4:4、YUV 4:2:2、YUV 4:2:0 这几种常用的类型。其中最常用的又是 YUV 4:2:0。这三种类型的 YUV 主要的区别就是 U、V 分量像素点的个数和采集方式。

YUV 4:4:4 就是每一个 Y 就对应一个 U 和一个 V；而 YUV 4:2:2 则是每两个 Y 共用一个 U、一个 V；YUV 4:2:0 则是每四个 Y 共用一个 U、V。我们可以通过图片来清晰地看一下三种 YUV 类型的区别。具体如下图所示：


每一个Y对应一个UV
YUV 4:4:4
表示Y
表示U
表示V
极客时间


每两个Y对应一个UV
YUV 4:2:2
表示Y
表示U
表示V
极客时间



总的来说：

1. YUV 4:4:4，每一个 Y 对应一组 UV。
2. YUV 4:2:2，每两个 Y 共用一组 UV。
3. YUV 4:2:0，每四个 Y 共用一组 UV。

YUV 4:4:4 这种类型非常简单，所以存储的方式也非常简单。

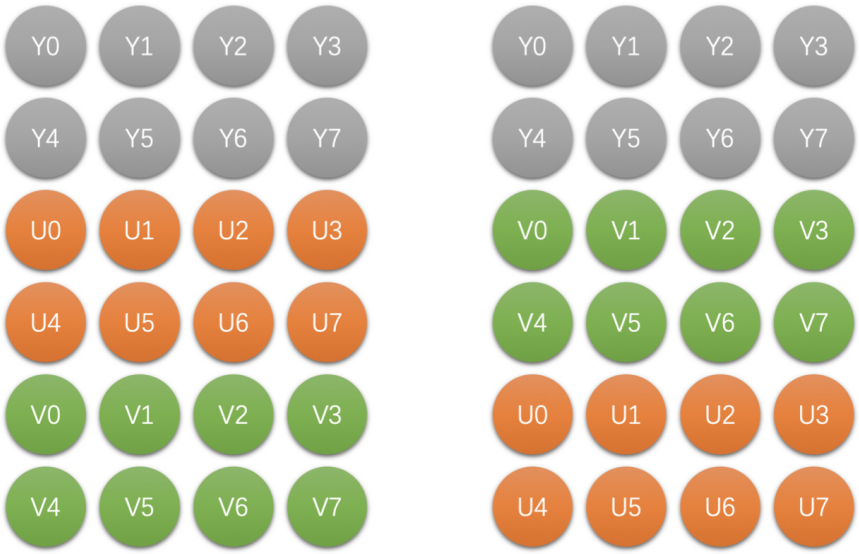
那 YUV 4:2:2 和 YUV 4:2:0 这种共用 U、V 分量的情况，应该在内存中怎么存储呢？下面我就来为你介绍一下。

YUV 存储方式主要分为两大类：Planar 和 Packed 两种。Planar 格式的 YUV 是先连续存储所有像素点的 Y，然后接着存储所有像素点的 U，之后再存储所有像素点的 V，也可以是先连续存储所有像素点的 Y，然后接着存储所有像素点的 V，之后再存储所有像素点的 U。Packed 格式的 YUV 是先存储完所有像素的 Y，然后 U、V 连续的交错存储。

下面我们就来看看每一种 YUV 类型的存储方式是怎么样的。

1. YUV 4:4:4

这种类型的 YUV 非常简单，因为每一个 Y 对应一个 U、一个 V，所以存储的方式也非常简单。例如，4 x 2 像素的 YUV 4:4:4 存储图如下图所示：



I444（YUV444）　　YV24（YUV444）

极客时间

可以看到，YUV 4:4:4 和 RGB 图像存储之后的大小是一样的。如果是 8bit 图像，就是每一个像素点需要占用 3 个字节。

2. YUV 4:2:2

这种类型的 YUV 稍微复杂些，每左右两个像素的 Y 共用一个 U 和一个 V。存储方式主要有以下 4 种类型。

YU16（或者称为 I422、YUV422P）

该类型是 Planar 格式，先存储完 Y，再存储 U，之后存储 V。例如，4 x 2 像素的 YU16 存储图如下图所示：

YU16（I422、YUV422P）

极客时间

YV16（YUV422P）

该类型也是 Planar 格式，先存储完 Y，再存储 V，之后存储 U。例如，4 x 2 像素的 YV16 存储图如下图所示：

YV16（YUV422P）

极客时间

NV16（YUV422SP）

这种类型是 Packed 格式，先存储完 Y，之后 U、V 连续交错存储。例如，4 x 2 像素的 NV16 存储图如下图所示：

NV16（YUV422P）

极客时间

NV61（YUV422SP）

这种也是 Packed 格式，与 NV16 不同，这种格式是先存储完 Y，之后 V、U 连续交错存储。例如，4 x 2 像素的 NV61 存储图如下图所示：

NV61（YUV422SP）

极客时间

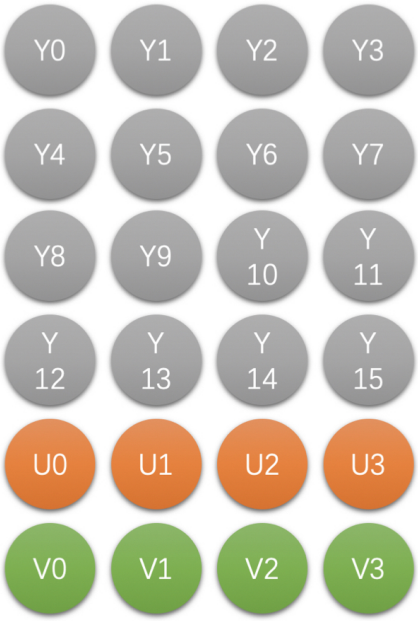
可以看到，4 x 2 像素的 YUV 4:2:2 只需要 16 个字节，而 RGB 图像则需要 24 个字节。也就是说，如果是 8bit 图像，那么 RGB 每一个像素需要 3 个字节，而 YUV 4:2:2 只需要 2 个字节。

3. YUV 4:2:0

这是最常见也是最常用的 YUV 类型。通常视频压缩都是 YUV 4:2:0 格式的。它是每上、下、左、右 4 个像素点共用一个 U 和一个 V。存储方式主要分为以下 4 种。

YU12（I420、YUV420P）

这种类型是 Planar 格式，先存储完 Y，再存储 U，之后存储 V。例如，4 x 4 像素的 YU12 存储图如下图所示：



YU12（I420、YUV420P）

极客时间

YV12（YUV420P）

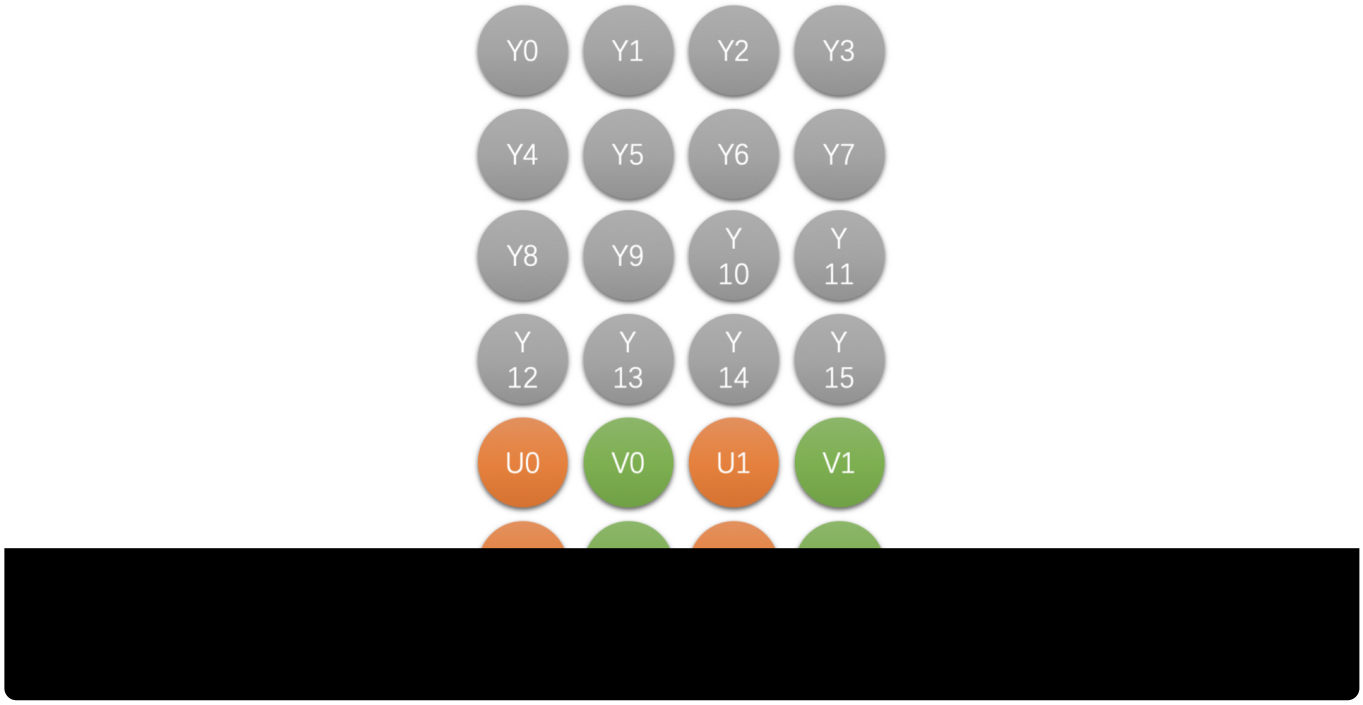
该类型也是 Planar 格式，先存储完 Y，再存储 V，之后存储 U。例如，4 x 4 像素的 YV 12 存储图如下图所示：

YV12（YUV420P）

极客时间

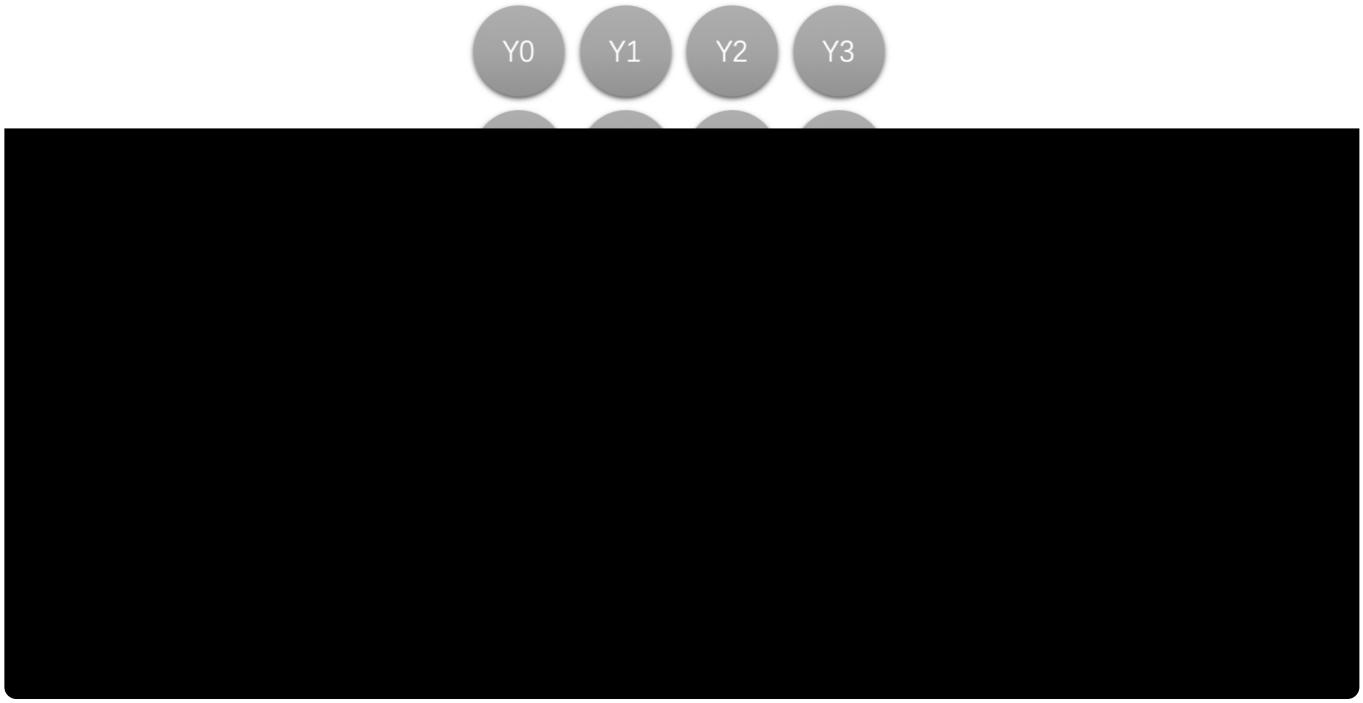
NV12（YUV420SP）

这种类型是 Packed 格式，先存储完 Y，之后 U、V 连续交错存储。例如，4 x 4 像素的 NV12 存储图如下图所示：



NV21（YUV420SP）

这种也是 Packed 格式，与 NV12 不同，这种格式是先存储完 Y，之后 V、U 连续交错存储。例如，4 x 4 像素的 NV21 存储图如下图所示：



可以看到，4 x 4 像素的 YUV 4:2:0 只需要 24 个字节相比 RGB 图像需要 48 个字节，存储的大小少了一半。也就是说，如果是 8bit 图像，RGB 每一个像素需要 3 个字节。而 YUV 4:2:0 只需要 1.5 个字节。

好了，以上就是 YUV 颜色空间的主要类型和存储方式。这里我用表格给你总结了一下。

YUV类型	存储类型	
YUV 4:4:4	I444	
	YV24	
YUV 4:2:2	Packed	NV16
		NV61
	Planar	YU16（I422）
		YV16
YUV 4:2:0	Packed	NV12
		NV21
	Planar	YU12（I420）
		YV12

极客时间

RGB 和 YUV 之间的转换

我们刚才说到，一般来说，采集到的原始图像、给显示器渲染的最终图像都是 RGB 图像，但是视频编码一般用的是 YUV 图像。那么这中间一定少不了两者的相互转换。那 RGB 如何转到 YUV 呢？YUV 又如何转到 RGB 呢？

在讲转换之前，我们先了解 Color Range 这个东西。对于一个 8bit 的 RGB 图像，它的每一个 R、G、B 分量的取值按理说就是 0~255 的。但是真的是这样的吗？其实不是的。这里就涉及到 Color Range 这个概念。Color Range 分为两种，一种是 Full Range，一种是 Limited Range。**Full Range 的 R、G、B 取值范围都是 0～255。而 Limited Range 的 R、G、B 取值范围是 16～235。**

了解了 Color Range 之后，我们怎么规范 YUV 和 RGB 之间的互转呢？其实这也是有标准的，目前的标准主要是 BT601 和 BT709（其实还有 BT2020，我们这里不展开讲）。简单来讲，**BT709 和 BT601 定义了一个 RGB 和 YUV 互转的标准规范**。只有我们都按照标准来做事，那么不同厂家生产出来的产品才能对接上。BT601 是标清的标准，而 BT709 是高清的标准。

下面我们来看看这两种标准分别在 Full Range 和 Limited Range 下的 RGB 和 YUV 之间的转换公式吧。具体如下图所示：

从上图我们可以看到每种标准下不同 Color Range 的转换公式是不同的。所以**在做 RGB 往 YUV 转换的时候我们需要知道是使用的哪个标准的哪种 Range 做的转换**，并告知对

方。这样对方使用同样的标准和 Range 才可以正确的将 YUV 转换到 RGB。

如果是系统采集出来给到用户的图像就是 YUV 的话，你也需要获取这个 YUV 的存储格式、转换标准和 Color Range。这样才能保证正确地处理 YUV 和 RGB 之间的转换。

好了，这就是关于 RGB 和 YUV 颜色空间的一些知识。在最后我需要再一次强调一下前一节课中讲到的 Stride。

在处理 YUV 图像的存储和读取的时候，也是有 Stride 这个概念的。事实上，YUV 出问题的情况更多。在这里举一个例子，比如说一张 1283x720 的图像，一个 Y 分量存储按 16 字节对齐的话应该是每行占用 1296 个字节，所以每读取一行像素的 Y 应该是 1296 个字节，具体如下图所示。千万不要认为是 1283 个字节，不然就会出现“花屏”。这里一定要注意。

小结

今天的知识点到这里就讲完了。总结一下，这节课我们主要讲了图像的颜色空间。图像的颜色空间主要有 RGB 和 YUV 两种。其中 RGB 图像每一个像素有 R、G、B 三个值。而 YUV 图像有 3 种类型，其分类如下：

1. YUV 4:4:4，一个 Y 对应一个 U 和一个 V；

2. YUV 4:2:2，左右两个 Y 对应一个 U 和一个 V；

3. YUV 4:2:0，上下左右四个 Y 对应一个 U 和一个 V。

同时，YUV 在存储的时候也有两种大类：一种是 Planar 格式；一种是 Packed 格式。其中 Planar 格式又分为先存 U 还是先存 V 两种。而 Packed 格式是 UV 交错存储且分为 U 在前还是 V 在前两种。

在图像采集的时候我们一般得到的原始图像是 RGB 图像，并且渲染的时候最终也是用 RGB 图像，而在编 / 解码时用的却是 YUV 图像。因此，我们需要在 RGB 和 YUV 之间互转。转换的标准有两种：一种是 BT601；一种是 BT709。

另外，在转换过程中我们还涉及到 Color Range 这个概念。Full Range 的 R、G、B 三个值的范围都是 0 ~ 255。而 Limited Range 的取值范围是 16 ~ 235。在做转换的时候我们需要选择正确的标准和 Color Range。

最后，我再一次提出了 Stride 这个概念。在读取 YUV 图像一行像素的时候一定要区分 Width 和 Stride 的区别。Width 是原始图像的宽，而 Stride 是对齐之后的一行像素使用的字节大小。这个一定要注意，千万不要弄错了。

思考题

通过这节课的学习，现在你知道我们为什么编码的时候使用 YUV 图像而不使用 RGB 图像了吗？

好了，这节课到这里就结束了，欢迎留言和我分享你的思考和疑惑，你也可以把今天所学分享给身边的朋友，邀请他加入探讨，共同进步。

分享给需要的人，Ta订阅后你可得 20 元现金奖励

生成海报并分享

赞 9 提建议

© 版权归极客邦科技所有，未经许可不得传播售卖。 页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 01 | 基本概念：从参数的角度看视频图像

下一篇 03 | 缩放算法：如何高质量地缩放图像？

精选留言

写留言

> 由作者筛选后的优质留言将会公开显示，欢迎踊跃留言。